

## CLAIMS

What is claimed is:

- 1 1. A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
  - 2 receiving a string of bits;
  - 3 generating a plurality of indices using a plurality of segments of bits in the string of bits;
  - 4 looking up simultaneously a plurality of entries from a plurality of look-up tables using the plurality of indices; and
  - 5 combining the plurality of entries into a first result;
  - 6 wherein the above operations are performed in response to the
  - 7 microprocessor receiving the single instruction.
- 1 2. A method as in claim 1 further comprising:
  - 2 receiving a plurality of data elements specifying the plurality of segments in the string of bits.
- 1 3. A method as in claim 2 wherein the plurality of data elements are received from an entry in a register file; and wherein the microprocessor is a media processor integrated with a memory controller on a single integrated circuit.

1    4.    A method as in claim 3 wherein the single instruction specifies an index of  
2                 the entry in the register file.

1    5.    A method as in claim 2 further comprising:  
2                 receiving a bit pointer, wherein the plurality of segments in the string of bits  
3                 are determined using the bit pointer and the plurality of data elements.

1    6.    A method as in claim 5 further comprising:  
2                 generating a new bit pointer using the first result.

1    7.    A method as in claim 1 further comprising:  
2                 receiving an offset, wherein the plurality of indices are determined using the  
3                 offset and the plurality of segments of bits.

1    8.    A method as in claim 1 further comprising:  
2                 partitioning look-up memory into the plurality of look-up tables before said  
3                 looking-up;  
4                 wherein the microprocessor is a media processor formed on a monolithic  
5                 integrated circuit.

1    9.    A method as in claim 8 wherein the look-up memory comprises a plurality of  
2                 look-up units, and wherein said partitioning look-up memory comprises:

3       configuring the plurality of look-up units into the plurality of look-up tables.

1     10.   A method as in claim 23 wherein each of the plurality of look-up units  
2           comprises 256 8-bit entries.

1     11.   A method as in claim 1 wherein the single instruction specifies a total  
2           number of entries contained in each of the plurality of look-up tables.

1     12.   A method as in claim 11 wherein the total number of entries is one of:  
2           a) 256;  
3           b) 512; and  
4           c) 1024.

1     13.   A method as in claim 1 wherein the single instruction specifies a total  
2           number of bits used by each entry contained in the plurality of look-up tables.

1     14.   A method as in claim 13 wherein the total number of bits is one of:  
2           a) 8;  
3           b) 16; and  
4           c) 24.

1     15.   A method as in claim 8 wherein the plurality of look-up tables are configured  
2           according to an indicator in an entry in a register file.

1    16. A method as in claim 15 wherein the single instruction specifies an index of  
2                         the entry in the register file.

1    17. A method as in claim 1 wherein said combining the plurality of entries  
2                         comprises:  
3                         selecting a valid data from the plurality of entries.

1    18. A method as in claim 17 further comprising:  
2                         generating an indicator indicating whether none of the plurality of entries is  
3                         valid.

1    19. A method as in claim 17 wherein the valid data is selected according to  
2                         priorities of the look-up tables from which the plurality of entries are looked  
3                         up.

1    20. A method as in claim 17 wherein said combining the plurality of entries  
2                         further comprises:  
3                         formatting the valid data according to a type of the valid data.

1    21. A method as in claim 20 wherein the type of the valid data is one of:  
2                         a) zero fill;  
3                         b) sign magnitude; and

- 4                   c) two complement.
- 1   22.   A method as in claim 21 further comprising:  
2                   retrieving a sign bit from the string of bits for the valid data, wherein the first  
3                   result is obtained by formatting the valid data using the sign bit when  
4                   the type of the valid data is sign magnitude.
- 1   23.   A method as in claim 1 wherein an entry in the plurality of entries contains:  
2                   a) information indicating whether the entry is valid;  
3                   b) information indicating a type of the entry; and  
4                   c) information indicating a number of bits of a code word to be decoded.
- 1   24.   A method as in claim 1 wherein the string is received from an entry in a  
2                   register file.
- 1   25.   A method as in claim 24 wherein the single instruction specifies an index of  
2                   the entry in the register file.
- 1   26.   A method as in claim 1 further comprising:  
2                   receiving a first number indicating a position of a last bit of input in the  
3                   string of bit.
- 1   27.   A method as in claim 26 further comprising:

2 generating an indicator indicating whether any bit after the last bit of input is  
3 used in obtaining the first result.

1 28. A method as in claim 12 further comprising:  
2 generating an indicator indicating whether one of the plurality of segments of  
3 bits contains a predetermined code.

1 29. A method as in claim 28 wherein the predetermined code represents an end  
2 of block condition.

1 30. A method as in claim 1 further comprising:  
2 receiving at least one format;  
3 formatting the string of bits into at least one escape data according to the at  
4 least one format; and  
5 combining the at least one escape data and the first result into a second result.

1 31. A method as in claim 30 wherein one of the at least one format is for data of  
2 a type which is one of:  
3 a) zero fill;  
4 b) sign magnitude; and  
5 c) two complement.

1 32. A method as in claim 30 wherein the at least one format is received from an  
2 entry of a register file.

1 33. A method as in claim 32 wherein the single instruction specifies an index of  
2 the entry in the register file.

1    34. A machine readable media containing an executable computer program  
2       instruction which when executed by a digital processing system causes said  
3       system to perform a method comprising:  
4           receiving a string of bits;  
5           generating a plurality of indices using a plurality of segments of bits in the  
6               string of bits;  
7           looking up simultaneously a plurality of entries from a plurality of look-up  
8               tables using the plurality of indices; and  
9           combining the plurality of entries into a first result;  
10          wherein the above operations are performed in response to the  
11               microprocessor receiving the single instruction.

1 35. A media as in claim 34 wherein the method further comprises:  
2 receiving a plurality of data elements specifying the plurality of segments in  
3 the string of bits.

- 1    36. A media as in claim 35 wherein the plurality of data elements are received  
2                         from an entry in a register file.
- 1    37. A media as in claim 36 wherein the single instruction specifies an index of  
2                         the entry in the register file.
- 1    38. A media as in claim 35 wherein the method further comprises:  
2                         receiving a bit pointer, wherein the plurality of segments in the string of bits  
3                         are determined using the bit pointer and the plurality of data elements.
- 1    39. A media as in claim 38 wherein the method further comprises:  
2                         generating a new bit pointer using the first result.
- 1    40. A media as in claim 34 wherein the method further comprises:  
2                         receiving an offset, wherein the plurality of indices are determined using the  
3                         offset and the plurality of segments of bits.
- 1    41. A media as in claim 34 wherein the method further comprises:  
2                         partitioning look-up memory into the plurality of look-up tables before said  
3                         looking-up.

- 1    42.    A media as in claim 41 wherein the look-up memory comprises a plurality of  
2                 look-up units, and wherein said partitioning look-up memory comprises:  
3                 configuring the plurality of look-up units into the plurality of look-up tables.
- 1    43.    A media as in claim 56 wherein each of the plurality of look-up units  
2                 comprises 256 8-bit entries.
- 1    44.    A media as in claim 34 wherein the single instruction specifies a total  
2                 number of entries contained in each of the plurality of look-up tables.
- 1    45.    A media as in claim 44 wherein the total number of entries is one of:  
2                 a) 256;  
3                 b) 512; and  
4                 c) 1024.
- 1    46.    A media as in claim 34 wherein the single instruction specifies a total  
2                 number of bits used by each entry contained in the plurality of look-up tables.
- 1    47.    A media as in claim 46 wherein the total number of bits is one of:  
2                 a) 8;  
3                 b) 16; and  
4                 c) 24.

1    48.    A media as in claim 41 wherein the plurality of look-up tables are configured  
2               according to an indicator in an entry in a register file.

1    49.    A media as in claim 48 wherein the single instruction specifies an index of  
2               the entry in the register file.

1    50.    A media as in claim 34 wherein said combining the plurality of entries  
2               comprises:  
3               selecting a valid data from the plurality of entries.

1    51.    A media as in claim 50 wherein the method further comprises:  
2               generating an indicator indicating whether none of the plurality of entries is  
3               valid.

1    52.    A media as in claim 50 wherein the valid data is selected according to  
2               priorities of the look-up tables from which the plurality of entries are looked  
3               up.

1    53.    A media as in claim 50 wherein said combining the plurality of entries  
2               further comprises:  
3               formatting the valid data according to a type of the valid data.

- 1    54. A media as in claim 53 wherein the type of the valid data is one of:
- 2            a) zero fill;
- 3            b) sign magnitude; and
- 4            c) two complement.
- 1    55. A media as in claim 54 wherein the method further comprises:  
2            retrieving a sign bit from the string of bits for the valid data, wherein the first  
3            result is obtained by formatting the valid data using the sign bit when  
4            the type of the valid data is sign magnitude.
- 1    56. A media as in claim 34 wherein an entry in the plurality of entries contains:  
2            a) information indicating whether the entry is valid;  
3            b) information indicating a type of the entry; and  
4            c) information indicating a number of bits of a code word to be decoded.
- 1    57. A media as in claim 34 wherein the string is received from an entry in a  
2            register file.
- 1    58. A media as in claim 57 wherein the single instruction specifies an index of  
2            the entry in the register file.
- 1    59. A media as in claim 34 wherein the method further comprises:

2 receiving a first number indicating a position of a last bit of input in the  
3 string of bit.

1 60. A media as in claim 59 wherein the method further comprises:  
2 generating an indicator indicating whether any bit after the last bit of input is  
3 used in obtaining the first result.

1 61. A media as in claim 45 wherein the method further comprises:  
2 generating an indicator indicating whether one of the plurality of segments of  
3 bits contains a predetermined code.

1 62. A media as in claim 61 wherein the predetermined code represents an end of  
2 block condition.

1 63. A media as in claim 34 wherein the method further comprises:  
2 receiving at least one format;  
3 formatting the string of bits into at least one escape data according to the at  
4 least one format; and  
5 combining the at least one escape data and the first result into a second result.

1 64. A media as in claim 63 wherein one of the at least one format is for data of a  
2 type which is one of:  
3 a) zero fill;

- 4           b) sign magnitude; and  
5           c) two complement.

1     65.   A media as in claim 63 wherein the at least one format is received from an  
2           entry of a register file.

1     66.   A media as in claim 65 wherein the single instruction specifies an index of  
2           the entry in the register file.